UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/804,268	03/13/2001	Makoto Muraishi	826.1697/JDH	9108

21171 7590 02/08/2007

STAAS & HALSEY LLP
SUITE 700
1201 NEW YORK AVENUE, N.W.
WASHINGTON, DC 20005

EXAMINER	
CHUONG, TRUC T	
ART UNIT	**PAPER NUMBER**
2179	

SHORTENED STATUTORY PERIOD OF RESPONSE	MAIL DATE	DELIVERY MODE
3 MONTHS	02/08/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

If NO period for reply is specified above, the maximum statutory period will apply and will expire 6 MONTHS from the mailing date of this communication.

Office Action Summary	Application No.	Applicant(s)	
	09/804,268	MURAISHI ET AL.	
	Examiner	Art Unit	
	Truc T. Chuong	2179	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *3* MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *14 November 2006*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1,10,11,14 and 18-25* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1, 10, 11, 14, and 18-25* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application
6) ☐ Other: _____.

DETAILED ACTION

This communication is responsive to the Amendment, filed 11/14/06.

Claims 1, 10, 11, 14, and 18-25 are pending in this application. In this communication, claims 1, 10, 11, and 14 are independent, and claims 2-9, 12-13, and 15-17 are cancelled. This action is made final.

The text of those sections of Title 35, U.S. Code not included in this action can be found in a prior office action.

Claim Rejections - 35 USC § 102

1. Claims 1, 10, 11, 14, and 18-25 are rejected under 35 U.S.C. 102(a) as being anticipated by JUnit (Screen Captures 1-33, from http://www.junit.org/, Publication Date can go back to January 01, 2000).

From Internet Browser (Internet Explorer or Netscape) → www.junit.org → a unit test framework known as JUnit (http://www.junit.org) automates the process of running these tests, letting you quickly see whether your program returns the results you expect. JUnit testing software makes the process of running unit tests very simple by providing support for JUnit. Once you have written a JUnit test class, you can simply choose the "Test Current Document" command from the Tools menu to run the tests and view the results. The name of the tests being run will be shown in the Test Output tab, with each test method turning green if it completes successfully and red if it fails. Without compiling the whole program (software) because the written software may contains errors, JUnit will automatically generate the objects, based on defined class and code instructions and code instructions of the program, such as screen, input

fields, command icons, search fields, data ranges etc. (see pages 12, 20, and 21). From the provided JUnit documents or web site www.junit.org, the document on pages 12 and 23 clearly show the publication date was February, 2000; moreover, the Applicants can easily find that the same JUnit document including features and functions of JUnit as presented in the final rejection if following the link http://www.junit.org/news/index.htm?start=121 of the same web site, which is clearly stated that Jtest with JUnit has been available since January 01, 2000 (see attached document was mailed along the Advisory Action dated July 24, 2006, titled "Automating and Improving Java Unit Testing: Using Jtest with JUnit").

As to claims 1 and 14, JUnit shows a test support apparatus for supporting a test of a screen program using a graphic user interface, comprising:

a test support class generation unit obtaining screen definition information defining a test target screen program that generates and controls a screen (JUnit, pages 12 and 20), and generating a test support class which is a subclass inheriting a class of the test target screen program responsive to the screen definition information (TestRunner reload all classes for each test run, page 2 and 10), and a class for testing the test target screen program (Without compiling the whole program (software) because the written software may contains errors, JUnit will automatically generate the objects, based on defined class and code instructions and code instructions of the program, such as screen, input fields, command icons, search fields, data ranges etc., e.g., pages 12 and 20, and page 21 shows fields and buttons can be simulated for testing);

a test specification generation unit generating a test specification for the test target screen program according to the definition information (JUnit will automatically generate the objects,

based on defined class and code instructions and code instructions of the program, such as screen, input fields, command icons, search fields, data ranges etc., e.g., pages 12 and 20), and providing the test specification for the test support class (e.g., input fields, command icons, search fields, data ranges etc., e.g., pages 12 and 20); and

a test execution unit conducting a test of the test target screen program defined by the screen definition information using the generated test support class to thereby test the screen program using the graphical user interface (JUnit will automatically generate the objects, based on defined class and code instructions and code instructions of the program, such as screen, input fields, command icons, search fields, data ranges etc., e.g., pages 12 and 20); and

a test data generation unit supporting input of input test data (JUnit, e.g., pages 1-12), by displaying on the screen a menu of a test data and its attribute according to the test specification, and embedding the test data instructed by an operator in an input field on the screen (e.g., pages 10, 12, and textual TestRunner and graphical TestRunner, page 4).

As to claim 10, this is a method claim of the apparatus claim 1. Note the rejection of claim 1 above.

As to claim 11, this is a computer program product claim of the apparatus claim 1. Note the rejections of claim 1 above.

As to claims 18 and 24, JUnit shows the apparatus according to claim 1, wherein

said test specification includes a test item and content of test related to the test data (e.g., input fields, command icons, search fields, data ranges etc., e.g., pages 12 and 20), the test related to the test data, the test item indicating whether the test data is a normal value or an

abnormal value (input fields, data ranges, e.g., page 12 and 20), and the content of test indicating the type of test item (Testing Key Widgets, e.g., page 13), and

said menu displayed on the screen includes the test item, the type of the test, and the test data (e.g., pages 12-13, and 20).

As to claims 19 and 25, JUnit teaches the apparatus according to claim 1, wherein

said test support class further deletes the test data executed by the test execution unit from the menu displayed on the screen (data from the input fields of pages 20 and 23 can be entered or deleted/removed with new input values).

As to claims 20-21, they are method claims of the apparatus claims 18-19. Note the rejections of claims 18-19 above respectively.

As to claims 22-23, they are computer program product claims of the apparatus claims 18-19. Note the rejections of claims 18-19 above respectively.

Response to Arguments

2. Applicant's arguments filed 11/14/06 have been fully considered but they are not persuasive.

Applicants argued and Examiner disagrees with the followings:

a. JUnit does not disclose "a test support class generation unit obtaining screen definition information defining a test target screen program that generates and controls a screen, and generating a test support class which is a subclass inheriting a class of the test target screen program responsive to the screen definition information, and a class for testing the test target screen program."

JUnit clearly shows how to generates and controls a screen (JUnit, pages 12 and 20), and TestRunner reloads all classes for each test run (page 2 and 10); then, from the GUI of JUnit, the generating process run without compiling the whole program (software) because the written software may contains errors. JUnit will automatically generate the objects, based on defined class and code instructions and code instructions of the program, such as screen, input fields, command icons, search fields, data ranges etc., (e.g., pages 12 and 20, and page 21 shows fields and buttons can be simulated for testing). In last the non-final communication, mailed 08/23/06, JUnit users can automate the test creation process and further boost software reliability with virtually no additional effort—by using Parasoft Jtest as well as JUnit. Jtest is an automated error prevention tool that complements and extends JUnit. When JUnit users add Jtest to their arsenal of tools, they can:

- Continue to run their existing JUnit test cases.
- Automatically generate new construction and functionality test cases.
- Automatically export Jtest test cases as JUnit test classes, and then add more test cases by modifying the test classes.
- Automatically create JUnit test class templates, and then add more test cases by modifying the templates.
- Automatically perform static analysis.
- Automatically increase and assess code coverage.

Essentially, by using Jtest and JUnit you streamline the unit testing process so that developers can *actually* perform comprehensive unit testing as often as they *intend* to perform comprehensive unit testing. The increased power that Jtest adds helps developers detect more errors in less time and prevent errors from occurring; this, in turn, leads to a shorter development cycle and a more reliable product. It clearly means the JUnit will automatically generate the objects, based on defined class, code instructions of the specification, and code instructions of the program, such as screen, input fields, command icons, search fields, data ranges etc. JUnit will automatically generate new construction and functionality test cases, automatically export Jtest test cases as JUnit test classes; then add more test cases by modifying the test classes, automatically create JUnit test class templates, and then add more test cases by modifying the templates. Other words, JUnit is considered in a same field with the invention, and JUnit also entirely covers the concept of claimed invention as explained above.

b. *JUnit does not teach or suggest "a test specification generation unit generating a test specification for the test target screen program according to the definition information, and providing the test specification for the test support class," and "a test execution unit conducting a test of the test target screen program defined by the screen definition information using the generated test support class to thereby test the screen program using the GUI."*

JUnit automatically generates the objects, based on defined class and code instructions and code instructions of the program, such as screen, input fields,

command icons, search fields, data ranges etc. (e.g., pages 12 and 20), and providing the test specification for the test support class (e.g., input fields, command icons, search fields, data ranges etc., e.g., pages 12 and 20); and JUnit can be used to generate new construction and functionality test cases, automatically export test cases and JUnit test classes, then add more test cases by modifying the test classes, automatically create JUnit test class templates, and then add more test cases by modifying the templates.

c. *JUnit does not show displaying on the screen a menu of a test data and its attribute according to the test specification, and embedding the test data instructed by an operator in an input field on the screen.*

From JUnit document (PARASOFT, see Advisory Action, mailed 07/24/06), JUnit clearly provides the menu of test class template (pages 3-4), the menu of test cases in UI (pages 5-6), and also shows the report menus of the test results (page 12). It clearly means the concept of claimed invention can be easily found throughout the provided JUnit document.

Conclusion

3. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after

the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Truc T. Chuong whose telephone number is 571-272-4134. The examiner can normally be reached on M-Th and alternate Fridays 8:30 AM - 5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Weilun Lo can be reached on (571) 272-4847. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

Truc T. Chuong

02/05/07

BA HUYNH
PRIMARY EXAMINER